Managing and Accessing 500 Million Files (and More!)

(Federating The NameSpace)

Mike Selway
Sr. Consulting Engineer

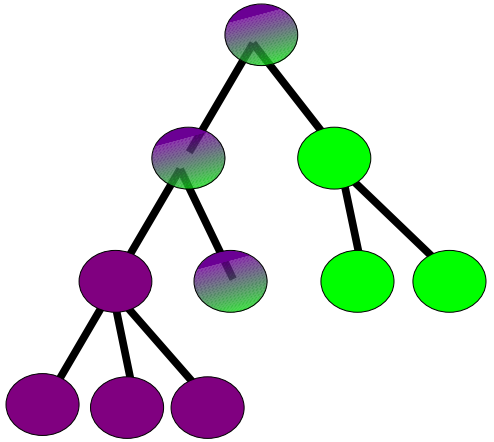
Sun Microsystems, Inc.
Mike.Selway@sun.com

Problem that is Growing: The Old and the New

- Asset Content is growing Exponentially
  - Legacy assets must be converted
  - New assets must be preserved
- Managing 1 million objects quite different to managing 500 million objects
- Cataloging file system (SAM-QFS) scales the common name space to billions of objects
  - "Future Proofing" Must be Designed Into Architecture
  - Planning for the Unknown Requires Massive Flexibility

Open Simple Access Hiding Complex Tiered Storage Management

- File Systems organize data
  - Oldest computer structure
  - Human Readable
  - Directories / Files
- All Computers have them
- Most data organized within them
- Therefore, the “best guess” for future proofing storage architectures is to create file system hierarchies

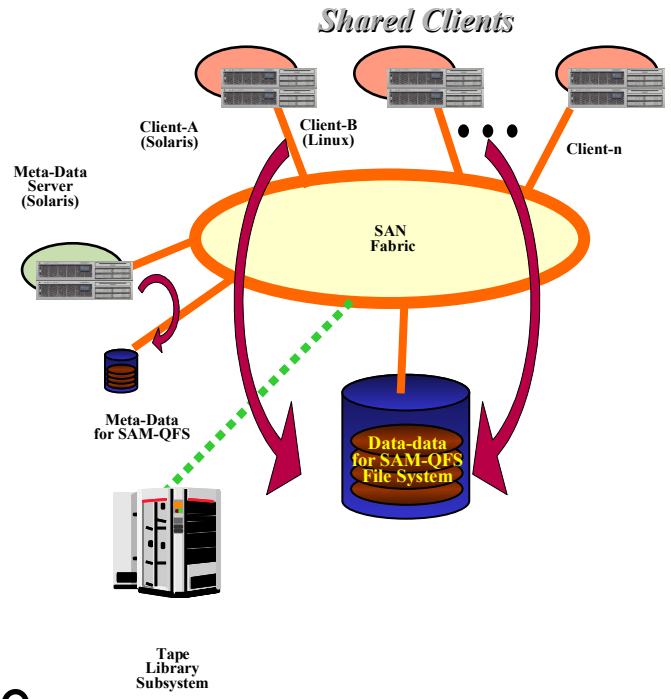


Open Simple Access Hiding Complex Tiered Storage Management

- Increasing importance of blending storage technologies
  - Cost / Energy efficient
  - Access efficient
- Sun Microsystems SAM-QFS software combines both requirements
  - Standard File System
  - Built-in Tiered Storage Management
  - Built in Future Proofing, File Integrity Checking, File Copy Validation

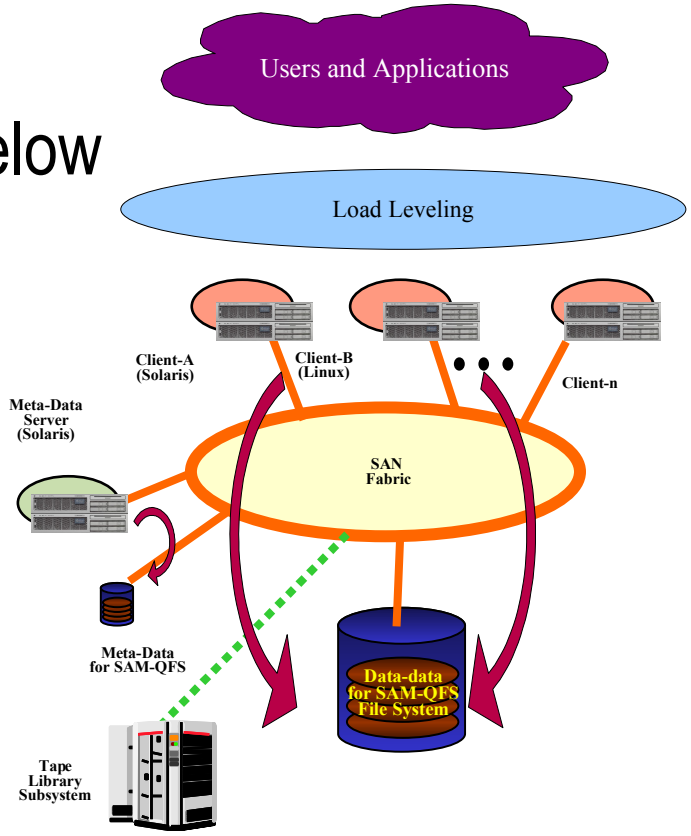
Expanding Requirements for Access to the System

- Performance / Scaling
  - > Metadata Server and “n” clients
  - > Sharing / dividing access
  - > Different clients, different purposes
- Control / Data Paths Separate
- Supports “Future Proofing”
  - > Access patterns will change
  - > Data access approaches will change
    - – equipment
    - – precision / resolution



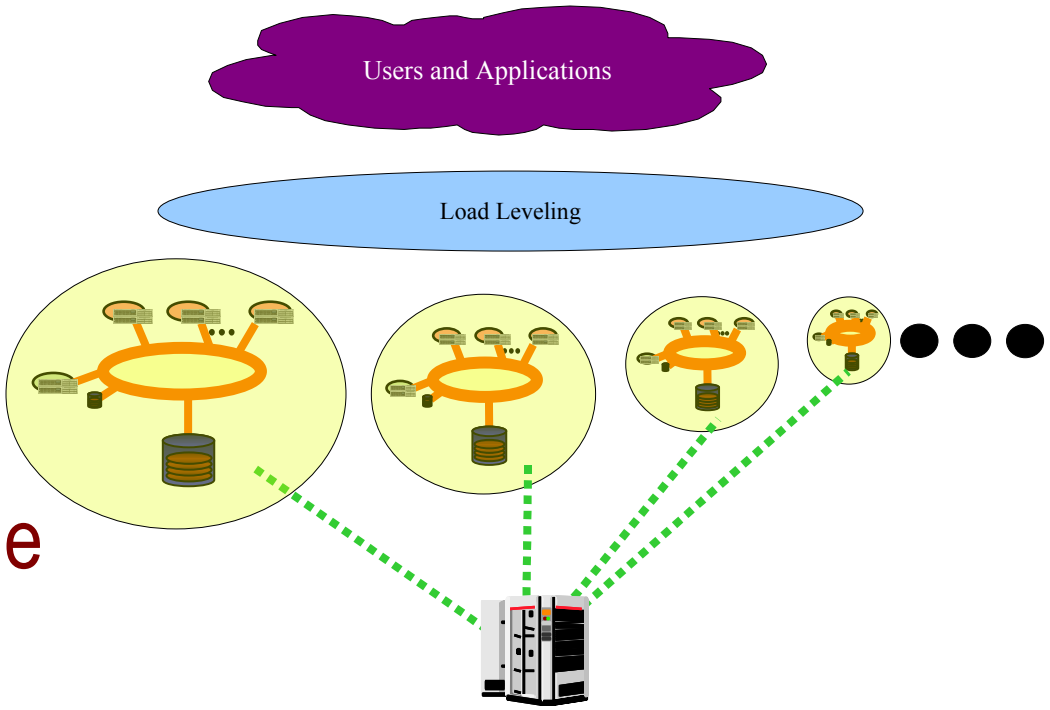
Expanding Requirements for Access to the System

- Establishing Common Access
  - > Virtualizing the Storage layers below
  - > Load Leveling Switches
  - > Cross-Platform Network Access
- Responsive to Growth
  - > Non-disruptive Changes
    - – Add new server types
    - – Remove old servers
    - – Add more servers for performance
  - > Built-in High Availability for failed server



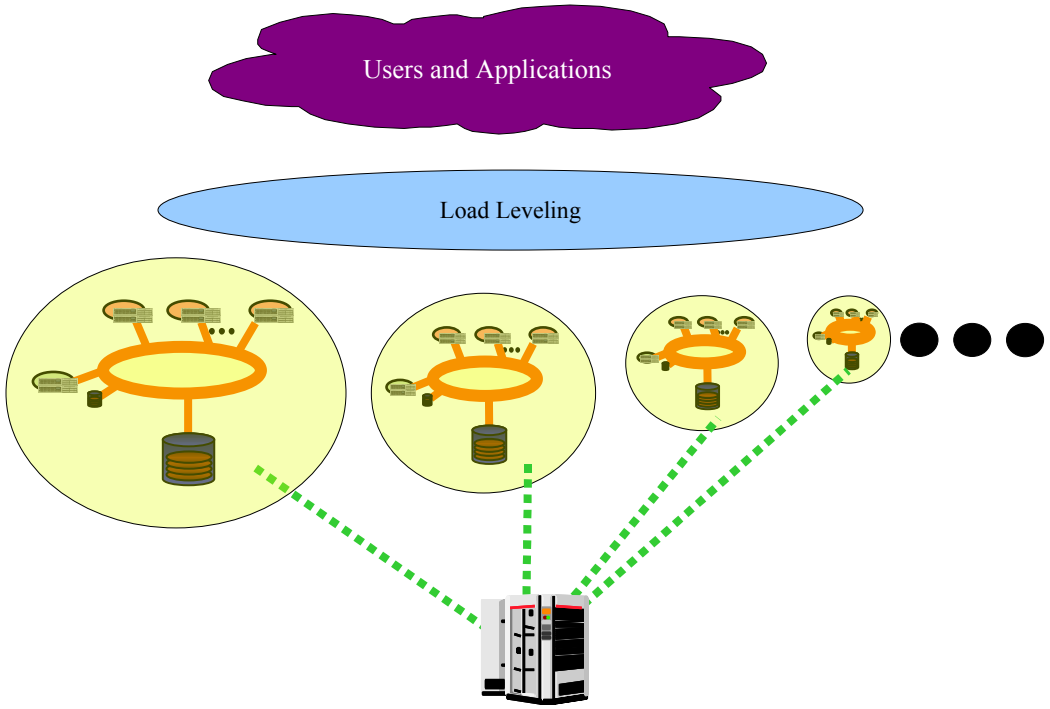
Scaling Beyond a Single Storage File System: Establishing a Common Name Space (Pods)

- Eventually, one file system not enough
  - > Numbers will hit "billions"
  - > Access to the files will evolve / expand
- Do *not* want to change the applications and access!
- Create a "module" of storage management



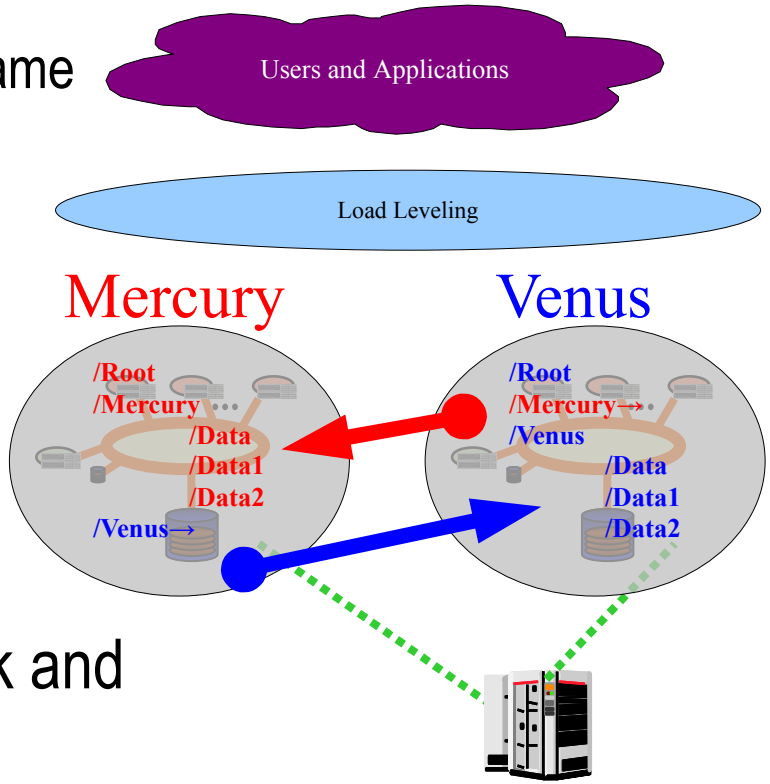
Scaling Beyond a Single Storage File System: Establishing a Common Name Space (Pods)

- Module Approach
  - Externally
    - set numbers of files, directory paths, number of users
  - Internally
    - number of clients, types of storage, types of storage
  - Pre-price Unit for rapid acquisition
  - Readily “link” different Pods together



Establishing a Common Name Space (Pods) for Billions of Files

- Federating the Name Space
  - > The name spaces link together to cooperatively form a single logical name space.
  - > Pods Link together via Standard UNIX protocols
    - – SymLinks
    - – When needed, also with NFS
- All Directories and Files Reachable from any Server
- Share Common Resources of disk and tape library
  - > Scale Horizontally by Pod and Add Links to other Pods
  - > Grow Groups of Pods into Cells



Assigning Policies to Data Management Based on Data Characteristics

- Scaling to Efficient Management (Storage and Access) of 100s of Millions of Files Into the Future
  - Crucial to Profile all Data
  - Some more important (today) than others
  - Use Storage Tiers to Maximize Value
  - “Ounce of Foresight Worth a Tonne of Cure”
- Get the Catalog Architected Right the First Time
- Let the File System Apply and Manage “the rules”
- Flexible to Change “the rules” as the System Evolves

Planning Ahead: Technology Refresh

- 100+ Year Archive Will Evolve Many Times
  - Every 3 - 5 years
  - New Storage, Tape, Management Technologies
  - New kinds of data, new kinds of users
- Massive Data Movements Required
  - Each iteration has “n” times more data to move
  - Compatibility of storage types impact ability to move
  - Locations for data may change
- SAM-QFS file system view
  - “Hides” the migrations
  - Future Proof Storage Layout (“tar”)
  - Upgrades Become Evolutionary, Not Revolutionary

Summary and Key Points

- File Systems are today's best architecture for long term asset cataloging and management
- SAM-QFS provides transparent storage management across multiple tiers
- Plan for Billions of files
- Future Proofing must be designed in from the beginning
- Plan for Technology Refreshing every 3-5 years
- Keep all decisions for asset organization and structure “flexible”

Managing and Accessing 500 Million Files (and More!)

(Federating The NameSpace)

Mike Selway
Sr. Consulting Engineer

Sun Microsystems, Inc.
Mike.Selway@sun.com